

Esame di Calcolatori Elettronici T

7 Luglio 2016 (Ing. Informatica)

Esercizio 1

Progettare un sistema basato sul microprocessore DLX dotato di **160 MB di EPROM** mappata negli indirizzi bassi e **136 MB di RAM** mappata negli indirizzi alti. Nel sistema sono presenti anche due periferiche a 8 bit, già progettate, denominate **INPUT_PORT**, in grado di ricevere dati dall'esterno utilizzando il protocollo di *handshake*, e **OUTPUT_PORT**, in grado di inviare dati all'esterno utilizzando il protocollo di *handshake*. In entrambi i casi si assuma che i dati trasferiti siano codificati con 8 bit senza segno. **L'interrupt generato da INPUT_PORT, prioritario rispetto a quello generato da OUTPUT_PORT, deve limitarsi a scrivere il dato letto in memoria all'indirizzo FFFF0000h.** **Per ogni trasferimento inerente OUTPUT_PORT: un'opportuna rete logica da progettare, deve sommare (modulo 256) il dato inviato dalla CPU** (si assuma a tal proposito che tale valore sia contenuto del registro R19) **con l'ultimo dato letto da INPUT_PORT solo se quest'ultimo è inferiore a 128.** A tal proposito, **si predisponga il sistema in modo che all'avvio l'ultimo dato letto risulti essere 255.**

- Progettare la rete logica che consente di sommare automaticamente, se necessario, al dato trasferito dall'interrupt handler verso OUTPUT_PORT l'ultimo dato letto da INPUT_PORT in un unico ciclo di bus. **Una soluzione puramente software a questo problema NON sarà considerata valida**
- Evidenziare e risolvere eventuali criticità
- Indicare i segnali di chip select di tutte le periferiche, memorie e di eventuali altri segnali presenti nel sistema
- Scrivere il codice dell'*interrupt handler* che gestisce le due interruzioni. Si assuma che i registri da R20 a R30 possano essere utilizzati senza la necessità di dover preservare il loro contenuto

Esercizio 2

Che cosa contiene il registro IAR nel DLX sequenziale? Per quale finalità è utilizzato? *Pc dell'indirizzo di ritorno dell'handler*

~~Esercizio 3~~

~~Spiegare per quale motivo è più efficace utilizzare nel Branch Target Buffer un meccanismo di predizione basato su due bit rispetto a una soluzione con un singolo bit.~~

MAPPING

EPROM 160 MB : EPROM 128 MB : $0 \times 0000\ 0000 \rightarrow 0 \times 07\text{FFFF}\text{FFFF}$

EPROM 32 MB : $0 \times 0800\ 0000 \rightarrow 0 \times 09\text{FFFF}\text{FFFF}$

RAM 136 MB : RAM 8 MB : $0 \times \text{F7}800000 \rightarrow 0 \times \text{F7}\text{FF}\text{FFFF}$

RAM 128 MB : $0 \times \text{F8}000000 \rightarrow 0 \times \text{FF}\text{FF}\text{FFFF}$

$\text{CS_RAM_8MB_0} = \overline{\text{BA31}} \overline{\text{BA27}} \text{BE0}$ $\text{CS_RAM_128MB_0} = \overline{\text{BA31}} \overline{\text{BA27}} \text{BE0}$
 $\text{CS_RAM_8MB_1} = \overline{\text{BA31}} \overline{\text{BA27}} \text{BE1}$ $\text{CS_RAM_128MB_1} = \overline{\text{BA31}} \overline{\text{BA27}} \text{BE1}$
 $\text{CS_RAM_8MB_2} = \overline{\text{BA31}} \overline{\text{BA27}} \text{BE2}$ $\text{CS_RAM_128MB_2} = \overline{\text{BA31}} \overline{\text{BA27}} \text{BE2}$
 $\text{CS_RAM_8MB_3} = \overline{\text{BA31}} \overline{\text{BA27}} \text{BE3}$ $\text{CS_RAM_128MB_3} = \overline{\text{BA31}} \overline{\text{BA27}} \text{BE3}$

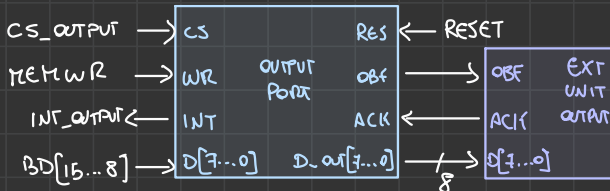
$\text{CS_EPROM_128_0} = \overline{\text{BA31}} \overline{\text{BA30}} \overline{\text{BA27}} \text{B60}$ $\text{CS_EPROM_32MB_0} = \overline{\text{BA31}} \overline{\text{BA30}} \overline{\text{BA27}} \text{B60}$
 $\text{CS_EPROM_128_1} = \overline{\text{BA31}} \overline{\text{BA30}} \overline{\text{BA27}} \text{B61}$ $\text{CS_EPROM_32MB_1} = \overline{\text{BA31}} \overline{\text{BA30}} \overline{\text{BA27}} \text{B61}$
 $\text{CS_EPROM_128_2} = \overline{\text{BA31}} \overline{\text{BA30}} \overline{\text{BA27}} \text{B62}$ $\text{CS_EPROM_32MB_2} = \overline{\text{BA31}} \overline{\text{BA30}} \overline{\text{BA27}} \text{B62}$
 $\text{CS_EPROM_128_3} = \overline{\text{BA31}} \overline{\text{BA30}} \overline{\text{BA27}} \text{B63}$ $\text{CS_EPROM_32MB_3} = \overline{\text{BA31}} \overline{\text{BA30}} \overline{\text{BA27}} \text{B63}$

$\text{CS_INPUT} = \overline{\text{BA31}} \overline{\text{BA30}} \text{1BF BE0}$

$\text{CS_OUTPUT} = \overline{\text{BA31}} \overline{\text{BA30}} \text{0BF BE1}$

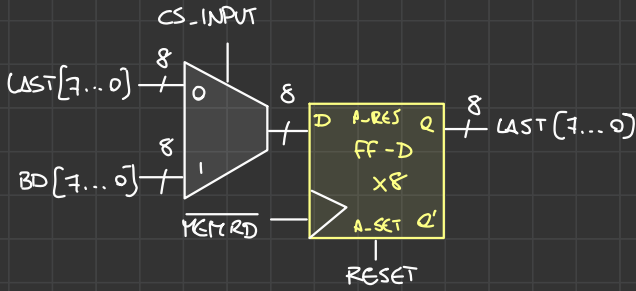
$\text{CS_READ_STATUS} = \overline{\text{BA31}} \overline{\text{BA30}} \text{BE2 MEMRD}$

PORTS I/O

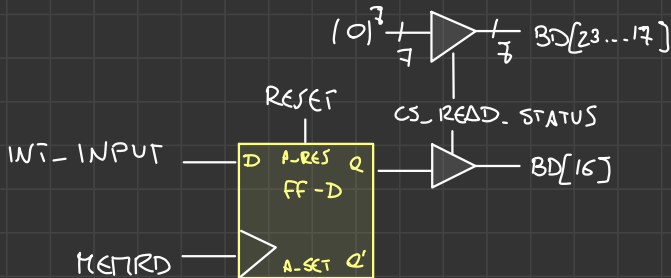
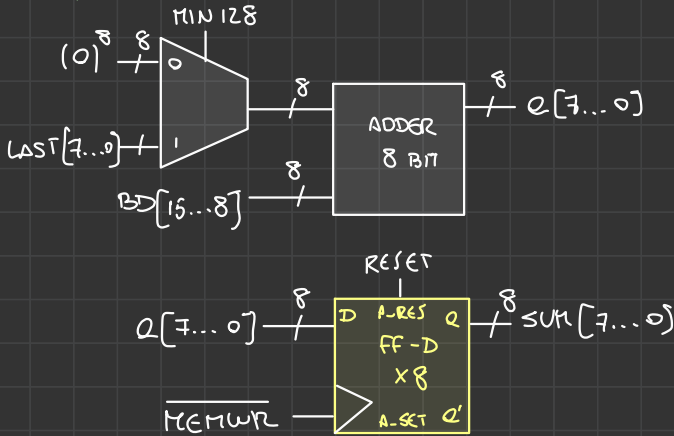


$\text{INT_TO_DLX} = \text{INT_INPUT} + \text{INT_OUTPUT}$

Registro Q' ultimo dato in INPUT



Rete per Q2 somma



CODICE

```
0h  LWI    R20, 0x4000
4h  LBU    R21, 0x0002(R20)
8h  BEQZ   R21, OUTPUT
Ch  LWI    R22, 0xFFFF
10h LBU    R23, 0x0000(R20)
14h SB     R23, 0x0000(R22)
18h  RFE
1Ch  OUTPUT SB  R19, 0x0001(R20)
20h      RFE
```

OUTPUT = 10h